

## AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

1. (currently amended) A method for creating a deterministic finite state automata (FSA) that match patterns in parallel, comprising:
  - creating states of the finite state automata from a set of patterns to be matched;
  - passing over the set of patterns a second time; and
  - adding transitions to the states that exactly matches to match all possible patterns that can start within the set of patterns to be matched.
2. (original) The method of claim 1 further comprising:
  - iterating through the states;
  - determining whether input causes a move to an initial state; and
  - if the initial state has a different move on the input, changing a current state's transition to mirror that of the initial state.
3. (currently amended) A method of creating a deterministic FSA that uses array-based transitions for an alphabet of size N, comprising:
  - representing each state as an object containing an array of N pointers to possible successive states;
  - using a numeric value of each member of the alphabet as an offset into the array to point to a next state.
4. Cancelled
5. (currently amended) A method for matching patterns in a deterministic FSA, comprising:
  - using a numeric value of less than a complete set of bits of an input as an offset into an array, thereby reducing a size of the array.
6. (original) The method of claim 5 comprising a further step of using a hash function for matching patterns composed of a 128 or 256 alphabet without overhead of larger arrays.

7. (new) The method of claim 1, wherein the step of adding transitions adds only those transitions that exactly match possible patterns that can start within the set of patterns to be matched.